

# RANCANG BANGUN SISTEM KENDALI OTOMATIS PINTU PAGAR BERBASIS MICROCONTROLLER NODEMCU ESP32 UNTUK KEAMANAN RUMAH

Jimmy<sup>1</sup>, Genrawan Hoendarto<sup>2</sup>, Jimmy Tjen<sup>3</sup>

Informatika Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak

<sup>1</sup>21421455\_jimmy@widyadharma.ac.id, <sup>2</sup>genrawan@widyadharma.ac.id, <sup>3</sup>jimmy\_tjen@widyadharma.ac.id

## Abstract

*In the modern era, home security has become a critical concern, particularly regarding gate access control, which is still often performed manually. This manual approach increases the risk of human error, such as unintentionally leaving the gate open, leading to reduced efficiency and heightened security vulnerabilities. This research aims to design and develop an automated gate control system based on the NodeMCU ESP32 microcontroller, operable via an Android application and a fingerprint sensor. A descriptive method was employed, using a literature review as the basis for system and application design. The resulting system enables users to automatically open and close the gate, monitor its status in real-time, manage user access rights, and receive alerts through notifications and a buzzer when the gate remains open beyond a specified time limit. Test results indicate that the system operates effectively, offering a secure, efficient, and practical solution for remotely managing home gate access.*

**Keywords:** Internet of Things, Automatic Control System, Real-time Monitoring, Fingerprint Sensor, User Access Management.

## Abstrak

Pada era modern, keamanan rumah menjadi aspek penting yang perlu diperhatikan, terutama dalam hal pengendalian akses pintu pagar yang masih banyak dilakukan secara manual. Kondisi ini berisiko menimbulkan kelalaian, seperti pintu pagar yang terbuka tanpa disadari, sehingga dapat mengurangi efisiensi dan meningkatkan potensi ancaman keamanan. Penelitian ini bertujuan untuk merancang dan membangun sistem kendali otomatis pintu pagar berbasis *microcontroller* NodeMCU ESP32 yang dapat dikendalikan melalui aplikasi Android dan sensor sidik jari. Metode yang digunakan adalah metode deskriptif, dengan pendekatan studi literatur sebagai dasar dalam perancangan sistem dan aplikasi. Sistem yang dikembangkan memungkinkan pengguna untuk membuka dan menutup pintu pagar secara otomatis, memantau kondisi pagar secara *real-time*, mengelola hak akses pengguna, serta menerima peringatan melalui notifikasi dan *buzzer* ketika pintu dibiarakan terbuka melebihi waktu yang ditentukan. Hasil pengujian menunjukkan bahwa sistem bekerja dengan baik dan memberikan solusi yang efisien, aman, dan praktis dalam pengelolaan pintu pagar rumah secara jarak jauh.

**Kata Kunci:** Internet of Things, Sistem Kendali Otomatis, Pemantauan *Real-time*, Sensor Sidik Jari, Pengelolaan Akses Pengguna.

## 1. PENDAHULUAN

Pada era modern, teknologi *Internet of Things* telah mengubah cara orang untuk berinteraksi dengan elemen arsitektur dalam kehidupan sehari-hari. Semua elemen, mulai dari elemen arsitektur rumah tangga hingga perkantoran, dapat dihubungkan ke internet dan dikendalikan dari jarak jauh. Dengan demikian, penerapan IoT menjadi sangat relevan untuk meningkatkan efisiensi dan keamanan di berbagai sektor, termasuk dalam lingkungan masyarakat. Kehadiran IoT memungkinkan pengelolaan bangunan menjadi lebih cerdas dan terintegrasi, mengurangi intervensi manual dan meningkatkan kenyamanan pengguna. IoT telah menjadi bagian penting dalam konsep *smart home* dan *smart building* yang semakin berkembang. Sistem-sistem rumah tangga seperti pencahaayaan, pendingin udara, hingga akses pintu, kini dapat diakses hanya melalui perangkat seluler. Hal ini menunjukkan bahwa transformasi teknologi turut mempengaruhi pola hidup masyarakat modern dalam aspek keamanan dan efisiensi.

Pada rumah pribadi, salah satu masalah yang sering dihadapi adalah ketergantungan pada pengontrolan manual perangkat seperti pintu pagar. Dalam situasi tertentu, seperti saat penghuni sedang terburu-buru meninggalkan rumah, proses membuka dan menutup pintu pagar secara manual memerlukan waktu dan tenaga yang tidak sedikit. Selain itu, situasi ini meningkatkan risiko lupa mengunci pintu pagar, yang mengakibatkan kurangnya efisiensi dan keamanan rumah. Masalah lainnya adalah kurangnya sistem pemantauan dan pengendalian pintu pagar, sehingga penghuni kesulitan memastikan apakah pintu pagar benar-benar telah tertutup dan terkunci dengan sempurna. Kondisi ini menciptakan potensi kerawanan terhadap akses tidak sah dari luar. Maka, perlu dirancang sebuah sistem yang mampu memberikan kontrol dan pemantauan secara otomatis untuk menjamin keamanan rumah secara menyeluruh dan efisien.

Sebagai solusi, teknologi IoT menawarkan sistem yang mampu mengintegrasikan dan mengotomatisasi kendali pintu pagar dengan menggunakan *microcontroller* NodeMCU ESP32. NodeMCU ESP32 merupakan *microcontroller* hemat daya yang terintegrasi dengan koneksi Wi-Fi dan Bluetooth *dual-mode*, sehingga sangat cocok digunakan dalam proyek-proyek IoT. Sistem ini dapat dihubungkan dengan berbagai sensor dan perangkat input-output, serta memungkinkan pengguna mengendalikan perangkat secara jarak jauh melalui jaringan internet. Pengguna dapat membuka atau menutup pintu pagar menggunakan ponsel, tanpa perlu hadir secara fisik. Teknologi ini tidak hanya memungkinkan penghematan energi dan waktu, tetapi juga meningkatkan aspek keamanan serta kenyamanan dalam pengontrolan akses pintu pagar. Integrasi sistem ini mendukung penerapan otomatisasi bangunan yang terstruktur dan terjangkau secara teknologi.

Selain dapat dikendalikan melalui ponsel, sistem ini juga memanfaatkan perangkat IoT berupa sensor sidik jari untuk membuka dan menutup pintu pagar secara cepat dan aman. Sistem ini dilengkapi dengan fitur pengingat berupa suara *buzzer* yang aktif saat pintu pagar masih terbuka, serta notifikasi yang dikirim ke ponsel jika pintu pagar dibiarkan terbuka terlalu lama. Fitur ini membantu pengguna dalam pengawasan tanpa harus selalu berada di dekat pintu pagar. Integrasi teknologi ini juga mendukung fitur log aktivitas dan manajemen akses pengguna, sehingga pemilik rumah dapat melihat siapa saja yang membuka pagar dan kapan. Dengan demikian, sistem ini mampu memberikan tingkat keamanan yang lebih tinggi dan efisiensi dalam penggunaan sehari-hari.

## 2. METODE PENELITIAN

Dalam penyusunan jurnal, penulis menggunakan rancangan penelitian deskriptif, di mana pada penelitian ini akan melibatkan analisis tentang bagaimana teknologi IoT dapat diterapkan untuk mengendalikan pintu pagar untuk efisiensi keamanan rumah.

### 2.1 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

#### a. Metode Wawancara

Metode wawancara melibatkan interaksi dengan pengguna yang masih menggunakan sistem kendali konvensional. Wawancara dilakukan untuk mendapatkan informasi mendetail mengenai kebutuhan, masalah, dan pengalaman pengguna dalam mengelola pintu pagar melalui kendali konvensional.

#### b. Metode Observasi

Metode observasi melibatkan pengamatan secara langsung terhadap pengguna yang masih menggunakan sistem kendali konvensional. Observasi yang dilakukan bertujuan untuk menganalisis masalah yang sering terjadi seperti efisiensi waktu, kenyamanan, dan kepraktisan dalam pengendalian perangkat.

#### c. Metode Studi Kepustakaan

Metode studi kepustakaan mencakup penelusuran berbagai literatur terkait, seperti buku, *e-book*, dan sumber informasi relevan lainnya yang berkaitan dengan sistem kendali otomatis pintu pagar menggunakan *microcontroller* NodeMCU ESP32. Melalui rujukan dari penelitian-penelitian sebelumnya, peneliti dapat membangun landasan teori serta memahami praktik nyata dalam merancang sistem kendali otomatis berbasis *microcontroller* untuk pintu pagar.

### 2.2 Teknik Analisis Sistem

Teknik analisis sistem yang digunakan dalam penelitian ini adalah *Unified Modeling Language* (UML). Dengan menggunakan teknis analisis data UML, penulis dapat memperoleh pemahaman yang lebih mendalam tentang sistem kendali otomatis pintu pagar yang sedang dianalisis dan dapat digunakan sebagai dasar acuan untuk merancang atau mengembangkan sistem kendali otomatis pintu pagar berbasis *microcontroller*.

### 2.3 Teknik Perancangan Sistem

Teknik perancangan sistem yang digunakan penulis dalam membangun sistem kendali otomatis pintu pagar adalah dengan menggunakan bahasa pemrograman C dengan Arduino IDE, perancangan *database* dengan Firebase, dan perancangan aplikasi dengan bahasa pemrograman Kotlin melalui Android Studio.

### 2.4 Landasan Teori

#### a. Sistem Kendali

Sistem kendali adalah suatu sistem yang terdiri dari beberapa elemen yang bertujuan untuk melakukan pengendalian suatu proses untuk mendapatkan suatu besaran yang diinginkan<sup>[1]</sup>. Sistem kendali dapat dikelompokkan menjadi dua berdasarkan karakteristiknya, yaitu sistem kontrol *loop* terbuka dan sistem kontrol *loop* tertutup. sistem kontrol *loop* terbuka merupakan suatu sistem kendali tanpa *feedback* (umpan balik), di mana nilai keluaran tidak memengaruhi aksi kontrol, sedangkan kontrol *loop* tertutup merupakan sistem kendali di mana nilai dari keluaran berpengaruh terhadap aksi sistem tersebut, identik dengan sistem kontrol umpan balik.

#### b. Internet of Things (IoT)

Internet of Things (IoT) adalah konsep di mana sekumpulan objek fisik atau perangkat lainnya dilengkapi dengan teknologi yang memungkinkan mereka untuk saling berkomunikasi melalui internet<sup>[2]</sup>. Salah satu peran dari IoT adalah dalam proses pengumpulan, penyimpanan, analisis, dan pengelolaan data secara efisien.

Perangkat IoT ini dapat berinteraksi dengan infrastruktur komputasi, baik pada pusat data maupun di lingkungan perangkat komputasi lainnya, termasuk perangkat *mobile*.

c. Sensor Sidik Jari

Sensor sidik jari adalah perangkat input biometrik yang digunakan untuk mengidentifikasi seseorang berdasarkan pola sidik jarinya<sup>[3]</sup>. Teknologi ini umumnya digunakan dalam sistem keamanan dan pengenalan identitas untuk meningkatkan keakuratan serta efisiensi dalam proses verifikasi pengguna.

d. Buzzer

*Buzzer* merupakan komponen elektronika yang dapat dipadukan pada papan Arduino. *Buzzer* biasanya digunakan sebagai alat untuk menghasilkan suara yang berperan sebagai pengingat atau alarm<sup>[4]</sup>. Dalam pengaplikasiannya, *buzzer* sering digunakan sebagai sebuah indikator suara untuk memberikan tanda pada suatu kondisi tertentu. Hal ini karena seseorang akan cenderung lebih awas terhadap suatu suara tertentu sebagai penanda kondisi tertentu dan lebih cepat menangkap informasi dari suara tersebut.

e. Motor Servo

Motor servo merupakan jenis motor yang mampu berputar dua arah, yaitu searah jarum jam (*Clockwise*) dan berlawanan arah jarum jam (*Counter-Clockwise*). Pergerakan motor servo dapat dikendalikan dengan memberikan sinyal PWM (*Pulse Width Modulation*) pada pin kontrolnya<sup>[5]</sup>. Terdapat dua tipe utama motor servo, yaitu servo *standard* yang berputar hingga 180 derajat dan servo *continuous* yang dapat berputar penuh hingga 360 derajat.

f. Magnetic Door Lock

*Magnetic Door Lock* adalah perangkat dengan sifat magnet yang sangat kuat yang dihasilkan secara elektromagnetik dan berfungsi sebagai kunci pintu. Apabila *Magnetic Door Lock* diberi tegangan maka akan menjadi magnet yang sangat kuat. *Magnetic Door Lock* menggunakan arus listrik sehingga menghasilkan gaya magnet yang lebih kuat. *Magnetic Door Lock* menggunakan gaya elektromagnetik untuk menghentikan pembukaan pintu, sehingga ideal untuk keamanan<sup>[6]</sup>. Sistem penguncian magnetik dapat dikontrol melalui aplikasi *mobile*.

g. Magnetic Reed Switch

*Magnetic Reed Switch* digunakan untuk mendeteksi setiap kali seseorang meninggalkan rumah, sensor pintu magnet digunakan untuk mendeteksi kapan pintu terbuka atau tertutup. Sensor yang digunakan untuk keperluan ini adalah *magnetic reed switch* yang dipasang pada pintu rumah<sup>[7]</sup>. Sensor ini bekerja dengan prinsip medan magnet yang memengaruhi saklar otomatis untuk mengontrol aliran arus listrik.

h. Mikrokontroler

Mikrokontroler adalah *single chip* komputer yang memiliki kemampuan untuk diprogram dan digunakan untuk melakukan tugas-tugas yang berorientasi kendali (*control*)<sup>[8]</sup>. Mikrokontroler dapat diprogram untuk menjalankan berbagai tugas yang berorientasi pada kendali serta mampu membaca dan menulis data sesuai dengan program yang telah ditetapkan.

i. NodeMCU ESP32

NodeMCU ESP 32 adalah serangkaian sistem hemat daya pada *microcontroller chip* dengan Wi-Fi terintegrasi dan Bluetooth *dual-mode* yang berguna untuk membuat sistem aplikasi IoT<sup>[9]</sup>. ESP 32 merupakan penerus dari *microcontroller* ESP8266 yang kompatibel dengan Arduino IDE.

j. Arduino

Arduino merupakan platform elektronik *open-source* yang dirancang untuk mempermudah pengembangan perangkat keras dan lunak. Dengan desain yang intuitif dan dapat meminimalkan kompleksitas teknis, Arduino mudah digunakan oleh berbagai kalangan, termasuk pemula<sup>[10]</sup>. Arduino memiliki fleksibilitas tinggi dalam implementasi berbagai proyek elektronik, mulai dari DIY sederhana hingga aplikasi industri yang kompleks, serta mampu menyimpan, mencatat, dan berbagi data dengan perangkat lain.

k. Bahasa Pemrograman Kotlin

Kotlin adalah bahasa pemrograman yang digunakan untuk pembuatan aplikasi Android, sehingga programmer yang memahami Kotlin dapat memanfaatkan peluang karir sebagai pembuat aplikasi Android<sup>[11]</sup>. Kotlin dianggap sebagai alternatif yang lebih mudah dipelajari dibandingkan Java, sehingga mempermudah proses pengembangan aplikasi dan memberikan peluang bagi programmer untuk berkariir di bidang pengembangan aplikasi Android.

l. Firebase

Firebase merupakan suatu layanan *Backend as a Service* (BaaS) yang dimiliki oleh Google dan digunakan untuk mempermudah para developer aplikasi dalam mengembangkan aplikasinya *mobile* maupun *web*<sup>[12]</sup>. Dengan itu, developer akan diper mudah dalam mengelola *backend* tanpa perlu membangun infrastruktur dari awal.

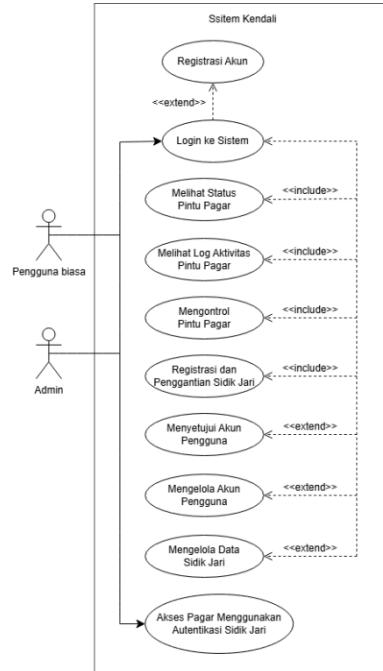
m. Unified Modeling Language (UML)

*Unified Modeling Language* (UML) merupakan bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya<sup>[13]</sup>. Selain itu, UML juga berperan dalam berbagai aspek pengembangan sistem, termasuk analisis kebutuhan, desain, implementasi, dan pengujian.

### 3. HASIL DAN PEMBAHASAN

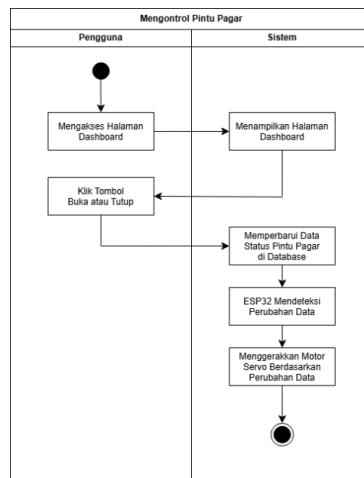
#### 3.1 Sistem Kendali Otomatis Pintu Pagar Berbasis Microcontroller

Sistem kendali otomatis pintu pagar yang dirancang berbasis *microcontroller* NodeMCU ESP32, berfungsi untuk mengendalikan pintu pagar melalui aplikasi atau sensor sidik jari. NodeMCU ESP32, sebagai pusat kendali utama memanfaatkan koneksi Wi-Fi untuk mengirimkan data menuju server atau *database*. Sistem ini dapat diakses oleh pengguna melalui aplikasi pada perangkat dengan sistem operasi Android, yang memungkinkan pengguna untuk melihat status pintu pagar, mengendalikan pintu pagar, serta mengelola data. Permodelan perancangan aplikasi yang akan dibuat disajikan menggunakan sistem berorientasi objek yaitu:



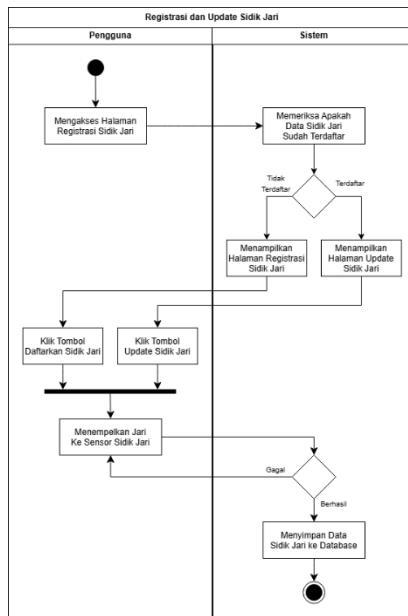
Gambar 1. Diagram Use Case Aplikasi

Diagram *use case* di atas menggambarkan rancangan sistem kendali otomatis pintu pagar berbasis *microcontroller* untuk efisiensi keamanan rumah. Sebelum dapat mengakses fitur-fitur di dalam sistem, pengguna perlu melakukan *login* terlebih dahulu. Setelah melakukan proses *login*, pengguna biasa memiliki akses terhadap fitur-fitur seperti mengontrol pintu pagar, melihat *log* aktivitas, melihat status pintu pagar. Sementara itu, admin memiliki akses tambahan untuk mengelola akun pengguna dan mengelola akun sidik jari.



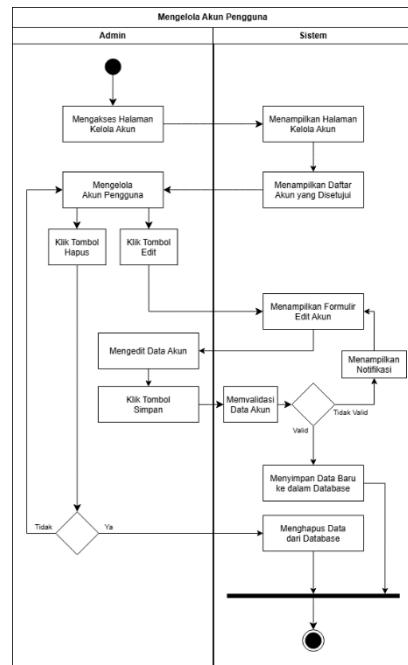
Gambar 2. Diagram Aktivitas Mengontrol Pintu Pagar

Cara kerja diagram aktivitas mengontrol pintu pagar pada Gambar 2 adalah pengguna terlebih dahulu mengakses halaman *dashboard* pada aplikasi. Pada halaman ini, pengguna dapat melihat status terkini pintu pagar serta memilih tombol "Buka" atau "Tutup" untuk memberikan perintah perubahan status. Setelah tombol ditekan, aplikasi akan memperbarui data status pintu pagar pada *database* secara *real-time*. NodeMCU ESP32 yang terus memantau perubahan data di *database* akan mendeteksi perintah tersebut dan segera mengaktifkan motor servo untuk membuka atau menutup pintu pagar sesuai perintah.



Gambar 3. Diagram Aktivitas Registrasi dan Update Sidik Jari

Cara kerja diagram aktivitas registrasi dan *update* sidik jari pada Gambar 3 adalah pengguna dapat mengakses halaman registrasi sidik jari. Sistem akan memeriksa apakah sidik jari pengguna sudah terdaftar. Apabila belum terdaftar, pengguna dapat klik tombol “Daftarkan Sidik Jari” dan mengikuti proses perekaman sidik jari dengan menempelkan jari ke sensor hingga berhasil. Setelah berhasil, data sidik jari akan disimpan ke dalam *database*. Namun, jika sidik jari sudah terdaftar, sistem akan mengarahkan pengguna ke halaman *update* sidik jari. Pada halaman ini, pengguna dapat memperbarui sidik jarinya dengan klik tombol “Update Sidik Jari” dan melakukan perekaman ulang melalui sensor hingga berhasil, kemudian data yang baru akan disimpan ke *database*.



Gambar 4. Diagram Aktivitas Mengelola Akun Pengguna

Cara kerja diagram aktivitas mengelola akun pengguna pada Gambar 4 adalah admin dapat mengakses halaman kelola akun. Kemudian, sistem akan menampilkan daftar akun yang disetujui. Pada halaman ini, admin dapat memilih akun untuk dikelola, seperti melakukan edit atau hapus. Apabila admin klik tombol “Edit”, sistem akan menampilkan formulir edit akun yang telah terisi dengan data akun terpilih. Admin dapat mengubah data tersebut, lalu klik tombol “Simpan”. Jika data valid, sistem akan memperbarui informasi di *database*. Jika tidak valid, sistem akan menampilkan notifikasi kesalahan dan tetap berada di halaman formulir untuk diperbaiki. Jika admin klik tombol “Hapus” dan mengkonfirmasi pilihan dengan klik tombol “Ya”, maka data akun akan dihapus dari *database*. Namun, jika memilih “Tidak”, sistem akan kembali menampilkan daftar akun untuk dipilih kembali.

### 3.2 Pengoperasian Aplikasi

Pengoperasian aplikasi kendali yang digunakan untuk mengendalikan pintu pagar melibatkan beberapa langkah, yaitu:

- Login

Pengguna diwajibkan untuk memiliki akun yang sudah diizinkan oleh admin dan telah diverifikasi melalui *email* sebelum dapat melakukan *login*. Pengguna dapat melakukan *login* dengan memasukkan *email* dan *password* yang dimiliki oleh pengguna sebelum dapat mengakses berbagai fitur yang terdapat di dalam aplikasi.

- Mengontrol Pintu Pagar

Pengguna dapat mengendalikan dan memantau status pintu pagar secara *real-time* melalui aplikasi. Dengan antarmuka yang sederhana, pengguna dapat dengan mudah menutup atau membuka pintu pagar.

- Registrasi dan Update Sidik Jari

Pengguna dapat mendaftarkan sidik jari melalui aplikasi dengan klik tombol “Daftar Sidik Jari”. Sistem akan memunculkan kotak dialog dan meminta pengguna untuk menempelkan jari sebanyak dua kali untuk verifikasi, kemudian menyimpan data ke sensor. Untuk pembaruan sidik jari, prosesnya sama seperti pendaftaran.

- Akses Pintu Pagar dengan Sidik Jari

Pengguna dapat mengakses pintu pagar menggunakan sensor sidik jari, namun sebelumnya harus melakukan proses registrasi sidik jari terlebih dahulu melalui aplikasi. Setelah terdaftar, pengguna cukup menempelkan jari pada sensor untuk membuka atau menutup pintu pagar secara otomatis.

- Logout

Setelah pengguna telah selesai menggunakan aplikasi. Fitur *logout* dapat digunakan untuk keluar dari sistem, sehingga penyalahgunaan aplikasi oleh pihak tidak dikenal dapat dihindari dan data tetap terlindungi dengan aman.

### 3.3 Tampilan Antar Pengguna

#### 3.3.1 Tampilan Halaman Dashboard



Gambar 5. Tampilan Halaman Dashboard

Pada Gambar 5 ditampilkan halaman *dashboard*. Pada halaman *dashboard*, pengguna dapat memantau status pintu pagar secara *real-time* dan melihat lima log aktivitas terbaru yang ditampilkan dalam bentuk tabel. Tersedia *button* untuk membuka dan menutup pagar. Di bagian kanan atas terdapat *button* untuk mengedit durasi pengiriman notifikasi saat pagar dibiarkan dalam kondisi terbuka. Sementara itu, di bagian kiri atas terdapat *button* menu yang berfungsi untuk menavigasikan pengguna berpindah ke halaman lain dalam aplikasi.

### 3.3.2 Tampilan Halaman Log Aktivitas

Nama	Aktivitas	Tanggal	Waktu
jimmy lim	Masuk ke sistem	24-06-2025	15:00:39
jimmy lim	Masuk ke sistem	24-06-2025	13:48:56
jimmy lim	Membuka Pagar dengan Sidik Jari	24-06-2025	02:49:14
jimmy lim	Menutup Pagar dengan Sidik Jari	24-06-2025	02:46:02
jimmy lim	Masuk ke sistem	23-06-2025	09:18:56
jimmy lim	Masuk ke sistem	23-06-2025	09:18:55
jimmy lim	Masuk ke sistem	23-06-2025	09:21:44
jimmy lim	Masuk ke sistem	23-06-2025	09:04:52
jimmy lim	Masuk ke sistem	23-06-2025	04:50:53

Gambar 6. Tampilan Halaman Log Aktivitas

Pada Gambar 6 ditampilkan halaman *log* aktivitas. Pada halaman *log* aktivitas, seluruh riwayat akses yang telah dilakukan oleh pengguna yang sedang *login* akan ditampilkan. Pengguna juga dapat memfilter data *log* berdasarkan rentang tanggal dengan mengisi tanggal awal dan akhir, kemudian klik button “TERAPKAN” untuk menampilkan hasil.

### 3.3.3 Tampilan Halaman Kelola Akun

Email	Nama	Role	Action
jimmylim247@gmail.com	jimmy lim	admin	

Gambar 7. Tampilan Halaman Kelola Akun

Pada Gambar 7 ditampilkan halaman kelola akun. Pada halaman kelola akun, *admin* dapat mengelola seluruh akun pengguna yang terdaftar dalam sistem. Terdapat button “PERMINTAAN AKSES LOGIN” yang dapat diklik untuk mengarahkan ke halaman permintaan akses dari pengguna baru. Selain itu, ditampilkan juga daftar akun pengguna yang telah terdaftar dan diizinkan mengakses sistem. *Admin* dapat melakukan pengeditan data akun pengguna, serta menghapus akun pengguna yang sudah tidak digunakan.

### 3.3.4 Tampilan Halaman Registrasi Sidik Jari

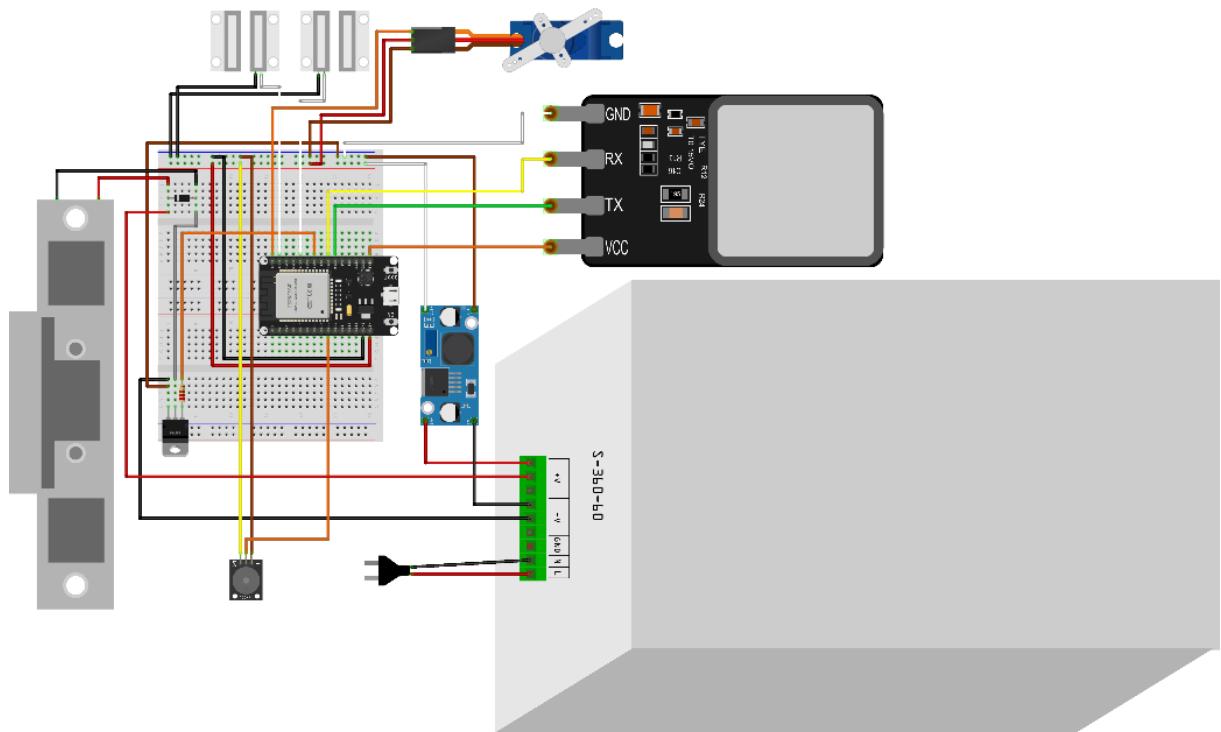


Gambar 8. Tampilan Halaman Registrasi Sidik Jari

Pada Gambar 8 ditampilkan halaman registrasi sidik jari. Pada halaman registrasi sidik jari, pengguna dapat klik button “DAFTARKAN SIDIK JARI” untuk memulai proses pendaftaran. Sistem akan menampilkan kotak dialog yang menginstruksikan pengguna untuk meletakkan jari pada sensor dan mengikuti proses pendaftaran hingga berhasil dilakukan. Jika proses berhasil, sidik jari akan terdaftar dan dapat digunakan untuk mengontrol pintu pagar melalui sensor sidik jari.

#### 3.4 Implementasi Perangkat Keras Sistem

Perangkat keras dibangun untuk membantu pengguna dalam mengendalikan pintu pagar secara *real-time* dan tidak dipengaruhi oleh jarak, yaitu dengan memanfaatkan sensor-sensor dan mengintegrasikannya dengan *microcontroller* NodeMCU ESP32. NodeMCU ESP32 berfungsi sebagai pusat kendali yang membaca data dari *database*.



Gambar 9. Rangkaian Sistem Kendali Otomatis Pintu Pagar

Tabel 1. Keterangan Warna Kabel

No.	Kabel	Keterangan
1.	Merah	Sebagai penghubung kutub positif catu daya ke perangkat-perangkat seperti <i>step-down LM2596</i> , <i>microcontroller NodeMCU ESP32</i> , <i>magnetic door lock</i> , motor servo MG996R, serta <i>power supply</i> 12V, untuk mengaktifkan dan menyuplai daya ke seluruh komponen tersebut.
2.	Hitam	Sebagai penghubung kutub negatif catu daya ke perangkat-perangkat seperti <i>step-down LM2596</i> , <i>microcontroller NodeMCU ESP32</i> , <i>magnetic door lock</i> , <i>power supply</i> 12V, serta <i>magnetic reed switch MC38</i> , untuk mengaktifkan dan menyuplai daya ke seluruh komponen tersebut.
3.	Coklat	Sebagai penghubung kutub negatif catu daya ke perangkat-perangkat seperti motor servo MG996R, <i>buzzer</i> 5V, serta pin <i>emitter</i> transistor NPN TIP120, untuk mengaktifkan dan menyuplai daya ke seluruh komponen tersebut.
4.	Abu-abu	Menghubungkan pin <i>collector</i> transistor NPN TIP120 menuju kutub negatif <i>magnetic door lock</i> .
5.	Oranye	Menghubungkan pin sinyal dari motor servo MG996R menuju pin GPIO 23 untuk komunikasi PWM, menghubungkan pin <i>base</i> dari transistor NPN TIP120 menuju pin GPIO 18 sebagai jalur kontrol, menghubungkan pin sinyal dari <i>buzzer</i> 5V menuju GPIO 27 dan untuk menyuplai tegangan 3.3V menuju pin VCC sensor sidik jari AS608 pada <i>microcontroller NodeMCU ESP32</i> .
6.	Putih	Menghubungkan pin sinyal <i>magnetic contact switch MC38</i> menuju pin GPIO 21 dan GPIO 22 untuk mendeteksi dan untuk menghubungkan pin GND sensor sidik jari AS608 menuju kutub negatif.
7.	Hijau	Menghubungkan pin TX dari sensor sidik jari AS608 menuju pin RX pada <i>microcontroller NodeMCU ESP32</i> untuk komunikasi serial (UART).
8.	Kuning	Menghubungkan pin RX dari sensor sidik jari AS608 menuju pin TX pada <i>microcontroller NodeMCU ESP32</i> untuk komunikasi serial (UART) dan untuk menyuplai pin VCC dari <i>buzzer</i> 5V menuju kutub positif.

Rangkaian kabel mengacu pada susunan atau tata letak kabel yang digunakan dalam sistem kendali otomatis pintu pagar. Kabel-kabel ini berfungsi untuk mengintegrasikan berbagai perangkat dengan mengalirkan listrik atau sinyal antara komponen-komponen yang berbeda dalam sistem.

### 3.5 Pengujian Aplikasi Terhadap Perangkat Keras

#### 3.5.1 Pengujian Otomatisasi Perangkat

Pengujian otomatisasi perangkat bertujuan untuk memastikan bahwa perangkat dapat mengaktifkan *buzzer* dan mengirimkan notifikasi ke aplikasi secara otomatis ketika pintu pagar terbuka selama waktu yang ditentukan.

Tabel 2. Pengujian Otomatisasi Perangkat

<b>Pengujian Otomatisasi Perangkat</b>				
No.	Aksi	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Mengaktifkan <i>buzzer</i> saat pagar terbuka selama 3 menit.	<i>Buzzer</i> aktif	<i>Buzzer</i> aktif ketika pagar terbuka selama 3 menit.	Berhasil
2.	Mengirimkan notifikasi ke aplikasi saat pagar terbuka melewati waktu notifikasi.	Notifikasi terkirim	Notifikasi terkirim ketika pagar terbuka selama waktu notifikasi yang ditentukan pengguna.	Berhasil

### 3.5.2 Pengujian Sistem Kendali

Pengujian sistem kendali bertujuan untuk memastikan bahwa semua fungsi utama sistem beroperasi dengan optimal dan tanpa hambatan.

Tabel 3. Pengujian Sistem Kendali

Pengujian Sistem Kendali				
No.	Aksi	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Membuka pintu pagar dengan aplikasi	Pintu pagar terbuka	Pintu pagar berhasil terbuka	Berhasil
2.	Menutup pintu pagar dengan aplikasi	Pintu pagar tertutup	Pintu pagar berhasil tertutup	Berhasil
3.	Membuka pintu pagar dengan sidik jari	Pintu pagar terbuka	Pintu pagar berhasil terbuka	Berhasil
4.	Menutup pintu pagar dengan sidik jari	Pintu pagar tertutup	Pintu pagar berhasil tertutup	Berhasil

### 3.5.3 Pengujian Terhadap Respon

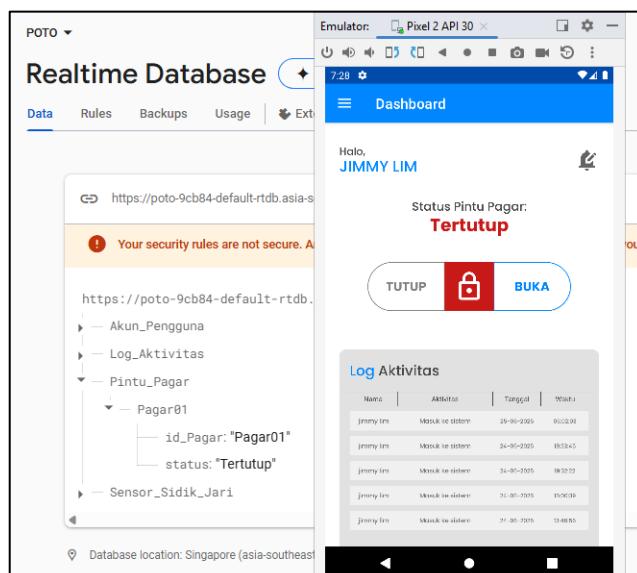
Pengujian respon bertujuan untuk mengetahui durasi yang dibutuhkan oleh perangkat untuk dapat merespon perintah yang diberikan pengguna melalui sistem kendali. Proses ini melibatkan pengiriman perintah menuju sensor dan pengukuran waktu yang dibutuhkan oleh sistem hingga perintah yang dikirimkan berhasil diterima dan dijalankan dengan baik oleh perangkat yang ingin dikendalikan.

Tabel 4. Pengujian Terhadap Respon

No.	Pengujian Yang Dilakukan	Respon (S)
1.	Button “BUKA” Pintu Pagar	1
2.	Button “TUTUP” Pintu Pagar	1
3.	Sensor Sidik Jari	2
4.	Magnetic door lock	2
5.	Otomatisasi Buzzer	2
6.	Otomatisasi Notifikasi	2

### 3.5.4 Pengujian Realtime Database

Pengujian realtime database bertujuan untuk memastikan bahwa data yang terlihat di dalam sistem telah sesuai dengan data yang tersimpan di firebase realtime database. Pengujian ini juga dilakukan untuk memverifikasi kecepatan sinkronisasi antara aplikasi dan database, sehingga perubahan data dapat ditampilkan secara akurat, tanpa keterlambatan secara real-time. Hal ini penting untuk menjaga konsistensi informasi kondisi perangkat..



Gambar 10. Perbandingan Data Status Pintu Pagar dengan Database

#### **4. KESIMPULAN**

Berdasarkan perancangan dan pengujian sistem kendali otomatis pintu pagar. Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

- a. Prototipe sistem kendali otomatis pintu pagar yang dirancang dapat mengontrol pintu pagar melalui sensor sidik jari serta mengontrol dan memantau secara *real-time* melalui aplikasi Android dengan baik.
- b. Berdasarkan pengujian menggunakan *blackbox testing*, sistem kendali otomatis pintu pagar dapat berfungsi dengan baik pada perangkat yang diuji. Sistem ini dapat membantu pengguna dalam mengontrol dan memantau pintu pagar secara *real-time* tanpa terpengaruh oleh jarak.
- c. Sistem secara otomatis mengirimkan notifikasi dan menyalakan *buzzer* ketika pintu pagar terbuka selama waktu yang ditentukan pada sistem kendali berfungsi dengan baik.

#### **5. SARAN**

Berdasarkan implementasi dan pengujian sistem kendali otomatis ruang kelas berbasis *microcontroller* NodeMCU ESP32 pada Universitas Widya Dharma Pontianak ini, penulis memberikan beberapa saran yang dapat digunakan untuk penelitian berikutnya, yaitu:

- a. Pada penelitian selanjutnya, disarankan agar sistem tidak hanya dikendalikan melalui aplikasi Android atau sensor sidik jari, tetapi juga dilengkapi dengan mekanisme pembuka manual sebagai solusi darurat saat terjadi pemadaman listrik. Selain itu, penggunaan UPS atau sistem cadangan daya dapat dipertimbangkan untuk menjaga fungsi dasar sistem agar akses ke pintu pagar tetap dapat dilakukan.
- b. Pada penelitian selanjutnya, disarankan agar aplikasi Android dapat mengirimkan notifikasi ketika pintu pagar terbuka selama waktu yang ditentukan, meskipun aplikasi sedang tidak dijalankan.
- c. Pada penelitian selanjutnya, disarankan agar sistem dilengkapi dengan LCD berbasis I2C untuk menampilkan informasi hasil interaksi dengan sensor sidik jari, seperti status autentikasi berhasil atau gagal. Penambahan ini bertujuan untuk memberikan umpan balik visual secara langsung kepada pengguna serta meningkatkan kemudahan penggunaan sistem.

#### **UCAPAN TERIMA KASIH**

Peneliti mengucapkan terima kasih yang sebesar-besarnya kepada civitas akademika Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak dan pihak-pihak lain yang memberikan dukungan dalam menyelesaikan penelitian ini dengan baik. Terima kasih.

## DAFTAR PUSTAKA

- [1] Carneiro, D., Navarro, E., Lloret, J., Chamoso, P., Novais, P., & Omatu, S. (2019). *Ambient Intelligence – Software and Applications –,10th International Symposium on Ambient Intelligence*. Berlin: Springer International Publishing.
- [2] Destriana, R., Husain, S. M., Handayani, N., & Siswanto, A. T. (2021). *Diagram UML Dalam Membuat Aplikasi Android Firebase "Studi Kasus Aplikasi Bank Sampah"*. Yogyakarta: Deepublish.
- [3] Erwin, E., Datya, A. I., Sepriano, S., Waryono, W., Adhicandra, I., Budihartono, E., & Purnawati, N. W. (2023). *Pengantar & Penerapan Internet Of Things: Konsep dasar & Penerapan IoT di berbagai Sektor*. Jambi: PT. Sonpedia Publishing Indonesia.
- [4] Husni, N. L., Caesarendra, W., Aznury, M., Novianti, L., & Stiawan, D. (2023). *Proceedings of the 6th FIRST 2022 International Conference (FIRST-ESCSI 2022)*. Paris: Atlantis Press International BV.
- [5] Ibadillah, A. F., & Alfita, R. (2021). *Mikrokontroler dan Aplikasinya*. Malang: Media Nusa Creative (MNC Publishing).
- [6] Mokoginta, D. (2024). *Pengantar Teknologi Informasi*. Batam: Cendikia Mulia Mandiri.
- [7] Mutaqin, R., & Yoziarde, L. A. (2025). *Pemrograman Arduino*. Bekasi: Pt Kimhsafi Alung Cipta.
- [8] Nugraha, J. A., Iwan, A. H., Gumuljo, E. F., Graciella, E., Mulyadinata, N., Budiman, K. O., & Nugroho, E. W. (2024). *Menguasai Arduino: Inspirasi Proyek-Proyek Arduino bagi Pemula*. Semarang: SIEGA Publisher.
- [9] Nugroho, G., & Norcahyo, R. (2024). *Mekatronika Pendekatan Praktis*. Depok: UGM PRESS.
- [10] Purnomo, R. F., Purbo, O. W., & Aziz, R. A. (2020). *Firebase Membangun Aplikasi Berbasis Android*. Yogyakarta: Andi Offset.
- [11] Setyawan, D. Y., Nurfiana, Syahputri, R., & Nurjoko. (2022). *Internet of Things ESP8266 ESP32 Web Server - Jejak Pustaka*. Yogyakarta: Jejak Pustaka.
- [12] Umam, F., Budiarto, H., & Dafid, A. (2021). *Motor Listrik*. Malang: Media Nusa Creative (MNC Publishing).
- [13] Wali, M., Sepriano, Nengsih, T. A., Hts, D. I., Antoni, Hasanuddin, Rachmat. (2023). *PENGANTAR 15 BAHASA PEMROGRAMAN TERBAIK DI MASA DEPAN (Referensi & Coding Untuk Pemula)*. Jambi: PT. Sonpedia Publishing Indonesia.